Vietnam J Comput Sci (2016) 3:93–102
DOI 10.1007/s40595-016-0060-6

REGULAR PAPER

Making kernel-based vector quantization robust and effective for incomplete educational data clustering

Thi Ngoc Chau Vo[1] · **Hua Phung Nguyen**[1] · **Thi Ngoc Tran Vo**[2]

Received: 3 December 2015 / Accepted: 19 February 2016 / Published online: 8 March 2016

Abstract Nowadays, knowledge discovered from educational data sets plays an important role in educational decision making support. One kind of such knowledge that enables us to get insights into our students' characteristics is cluster models generated by a clustering task. Each cluster model presents the groups of similar students by several aspects such as study performance, behavior, skill, etc. Many recent educational data clustering works used the existing algorithms like *k*-means, expectation–maximization, spectral clustering, etc. Nevertheless, none of them considered the incompleteness of the educational data gathered in an academic credit system although incomplete data handling was figured out well with several different general-purpose solutions. Unfortunately, early in-trouble student detection normally faces data incompleteness as we have collected and processed the study results of the second-, third-, and fourth-year students who have not yet accomplished the program as of that moment. In this situation, the clustering task becomes an inevitable incomplete educational data clustering task. Hence, our work focuses on an incomplete educational data clustering approach to the aforementioned task. Following kernel-based vector quantization, we define a robust effective simple solution, named VQ_fk_nps, which is able to not only handle ubiquitous data incompleteness in an iterative manner using the nearest prototype strategy but also optimize the clusters in the feature space to reach the resulting clusters with arbitrary shapes in the data space. As shown through the experimental results on real educational data sets, the clusters from our solution have better cluster quality as compared to some existing approaches.

Keywords Incomplete data clustering · Educational data mining · Kernel-based vector quantization · Nearest prototype strategy · Non-spherical cluster

✉ Thi Ngoc Chau Vo
chauvtn@cse.hcmut.edu.vn

Hua Phung Nguyen
phung@cse.hcmut.edu.vn

Thi Ngoc Tran Vo
vtntran@hcmut.edu.vn

[1] Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology, Vietnam National University, Ho Chi Minh City, Vietnam

[2] School of Industrial Management, Ho Chi Minh City University of Technology, Vietnam National University, Ho Chi Minh City, Vietnam

1 Introduction

Educational data mining is nowadays well known worldwide for discovering knowledge hidden in educational data to support educational decision making. As one of the widely used mining tasks, educational data clustering has been considered with many different student-related aspects in [4,5,11–13,15,18,20–22,26] for many various purposes. For example, discovering the groups of similar students is based on study performance in [11], learning behavior in [15], skill in [18], preference in [26], etc. A variety of data have been collected and processed for clustering the students. For example, Bogarín et al. [4] used data of the undergraduate students in an online course using Moodle and Zakrzewska [26] got data about the undergraduate and graduate students participating in the experiments in online collaboration also on Moodle, Refs. [5,11,20] clustered the data of undergraduate students and courses in a few years, Jayabal and Ramanathan [12] analyzed the 10th grade data, Kerr and Chung [13] extracted the student performance features from log data in educational video games and simulations, Li and Yoo [15] used the data recorded from each student's actual lab experi-

ence, and so on. Among these related works, only Inyang and Joshua [11] has presented the handling of incomplete data by deleting the missing results in the courses while the others had no mention of incomplete data issues. If the number of missing values is small, ignoring them might have no impact on the performance of the clustering task. Otherwise, the effectiveness of the clustering task might be influenced by data insufficiency. In addition, incomplete data are ubiquitous in the data gathered in an academic credit system, especially in the case we would like to provide any appropriate support to the students as soon as possible in the second, third, or fourth year of their study period. So, our work in this paper is dedicated to a clustering task on incomplete educational data.

Despite such a lack of incomplete data handling for an educational data clustering task, we are aware of many existing works on incomplete data clustering in general such as [1,2,7–9,23,25,27]. Among these works, Refs. [1,2,9,23,27] updated incomplete data while doing data clustering, Wang [25] translated incomplete data into so-called fuzzy observations before generating clusters, and Refs. [7,8] estimated incomplete data after attaining clusters. Thus, it can be seen that Wang [25] handled incomplete data in the preprocessing phase of the clustering task, Refs. [1,2,9,23,27] tackled incomplete data in the clustering phase of the clustering task, and Refs. [7,8] performed post-processing incomplete data after the clustering task was done. Furthermore, some existing works such as Refs. [9,27] have constrained the resulting shapes of the clusters. Fuzzy c-means, mean-shift clustering, and the self-organizing map (SOM) learning algorithm are some examples which have been enhanced for incomplete data clustering. Nonetheless, it is hard for us to foresee which incomplete data handling techniques are certainly appropriate for a particular application domain and also work well with any existing learning algorithms. Therefore, a study of handling incomplete data in a clustering task is needed to attain an effective cluster model in general and in the education domain.

Based on the motivations stated previously, we concentrate on a solution to the incomplete educational data clustering task in an academic credit system. In particular, we define a kernel-based vector quantization approach to effectively clustering incomplete educational data where incompleteness can be present in any data record at any dimension. Indeed, this work is an extended version of the work previously proposed in [24] where we focused on an effective algorithmic framework for incomplete educational data clustering and illustrated the applicability of the framework with the two proposed algorithms: K_nps and S_nps. K_nps followed the partitioning approach while S_nps did the neural network-based approach. Both of them utilized the nearest prototype strategy for incomplete data updates while do clustering within our proposed algorithmic framework. Like K_nps and S_nps, this work is also based on the proposed framework in [24]; however, defines a new approach using kernel-based vector quantization proposed in [10] and the nearest prototype strategy to handle the ubiquity of incomplete data. The resulting algorithm, named VQ_fk_nps, is our novel incomplete educational data clustering approach as compared to the algorithms K_nps and S_nps which have been achieved in [24]. As compared to K_nps and S_nps, VQ_fk_nps is capable of forming the groups of the similar students which are the clusters with non-spherical shapes in the data space corresponding to the spherical clusters in the feature space. Via the experimental results on real educational data sets with internal clustering validation measures, VQ_fk_nps is confirmed to outperform several existing approaches and thus, be effective for incomplete educational data clustering. Besides, the incomplete data sets that become completed after the data clustering task can be utilized in other mining tasks such as classification and association analysis.

2 Kernel-based vector quantization for incomplete educational data clustering

2.1 Incomplete educational data clustering task definition

To be self-contained, an incomplete educational data clustering task is re-stated in this subsection although first introduced in [24]. This task is a performance-based student clustering task in an academic credit system.

Our educational data mining is dedicated to regular undergraduate students being studying at a university using an academic credit system. During the period of study time, each student has a study status and might face many difficult problems that would influence their study and then make them fail to get a degree from the university. Therefore, we would like to support the current undergraduate students appropriately by examining the cases of the similar students in the past. This leads us to the necessity of grouping the students according to their similar study results, i.e. *a performance-based student clustering task* considered in this paper.

The input of the task is a set $\boldsymbol{D}$ of data vectors each of which represents a student. Dimensions of each vector correspond to the subjects each student has to successfully study to accomplish the program. A value of each vector at a dimension is a grade that the student gets after taking a subject. If the student has not yet taken a subject, its grade is not available and its value at the corresponding dimension is incomplete. Thus, the study performance of each student is reflected through the values of the corresponding vector. In general, $\boldsymbol{D}$ is specified with n data vectors in a p-dimensional data space: $\boldsymbol{D} = \{X_1, X_2, \ldots, X_n\}$ where $X_j = (x_{j,1}, x_{j,2}, \ldots, x_{j,p})$ for $j = 1, \ldots, n$.

The output of the task is a collection of clusters each of which has some similar data vectors. Indeed, each cluster rep-

resents a group of the students who have similar study performance. Therefore, they can be considered in the similar cases.

In practice, data gathered in a flexible academic credit system are incomplete. Such data incompleteness is the peculiarity that makes our task challenging. As discussed in [24], *incomplete data can exist in any vector at any dimension in the educational data sets archived within an academic credit system*. That is: $\forall \mathrm{j} = 1..n, \exists dim = 1..p, \mathrm{x}_{j,dim} \neq \mathrm{NULL}$ and $\forall dim = 1..p, \exists\ \mathrm{j} = 1..n, \mathrm{x}_{j,dim} \neq \mathrm{NULL}$ where NULL is an incomplete value in a given data set $\boldsymbol{D}$. This situation shows the ubiquity of incomplete data in the data we gathered and processed in order to perform a clustering task and thus, requires a clustering solution to pay attention to such data incompleteness.

Unfortunately, no existing educational data mining work has taken data incompleteness into account thoroughly except for [24]. In this paper, our work follows the kernel-based approach with vector quantization and the nearest prototype strategy by proposing a robust and effective incomplete educational data clustering algorithm, named VQ_fk_nps, where we tackle data incompleteness in the data space while doing clustering in the feature space. The reasons for choosing kernel-based vector quantization are given as follows. First, it uses a competitive learning mechanism for grouping the similar objects into several single clusters in the entire data space in a simple but intuitive way. Second, it follows a kernel-based approach that is capable of generating the resulting clusters with non-spherical shapes inherent in the data space corresponding to spherical clusters in the feature space. Finally, it is an iterative clustering algorithm that enables us to embed an incomplete data update phase in an elegant manner without breaking the original clustering procedure.

2.2 Making kernel-based vector quantization robust and effective for incomplete educational data clustering

In this section, a kernel-based approach with vector quantization and the nearest prototype strategy is proposed to effectively clustering incomplete educational data. The resulting algorithm is named VQ_fk_nps (*V*ector *Q*uantization for data clustering in a *F*eature space using the Gaussian *K*ernel function and the *N*earest *P*rototype *S*trategy).

The foundations of a robust kernel-based approach with vector quantization for effectively clustering incomplete educational data

The foundations of our approach are based on the framework proposed in [24] which is the generalization of OCSFCM and NPSFCM algorithms in [9]. This is because Hathaway and Bezdek [9] has taken into account the ubiquity of incomplete data in the same way as we found in the educational data sets earlier discussed in Sect. 2.1. Besides, Hathaway and Bezdek [9] defined a gentle objective-based approach for both clustering and incomplete data handling towards the best resulting clusters in an iterative manner. Moreover, Hathaway and Bezdek [9] has embedded the fuzzy c-means algorithm in steps 2–4 of OCSFCM algorithm completely in the original state in each epoch. These characteristics facilitate the task with any existing iterative algorithms in various clustering approaches.

As previously defined along with the framework in [24], K_nps is an incomplete data clustering version of the k-means algorithm and S_nps is the one of the self-organizing algorithm, both using the nearest prototype strategy. K_nps and S_nps have been tested with the generalization of OCSFCM and NPSFCM algorithms. Experimental results have shown that the proposed framework and its algorithms seem to be appropriate for clustering educational incomplete data sets.

Similarly, in this paper, which is an extended version of [24], VQ_fk_nps is proposed and also dedicated to the education domain. VQ_fk_nps is an incomplete data clustering version of kernel-based vector quantization using the nearest prototype strategy. The main difference between VQ_fk_nps and our previous incomplete educational data clustering algorithms, K_nps and S_nps, is the formulation of the clusters of the similar objects in the feature space with the Gaussian kernel function instead of that in the data space. This approach will enable us to find the non-spherical clusters truly based on the nature of the objects in the data space by means of the simple, efficient, and intuitive unsupervised learning procedure of vector quantization in the feature space. Therefore, we expect to achieve the resulting clusters of better quality in terms of compactness and separation. Besides, we generalize the incomplete data handling approach in the preprocessing phase as well as in the post-processing phase in the proposed algorithm. As a generalized version of the incomplete data handling approach in the preprocessing phase, VQ_fk_nps starts clustering the objects with data completeness after the initial updates on incomplete data using some existing data cleaning method such as attribute means. As a generalized version in the post-processing phase, VQ_fk_nps performs the updates on incomplete data using the nearest prototype strategy after the movement of the reference vectors closer to their members. Furthermore, incomplete data update is made in an iterative manner so that our incomplete data handling process is optimized along with the optimization of the clustering process for better clusters with an approximation of incomplete data.

VQ_fk_nps: a kernel-based vector quantization approach to incomplete data clustering using the nearest prototype strategy

In the following, we describe our incomplete educational data clustering algorithm, VQ_fk_nps, using kernel-based vec-

tor quantization proposed in [10]. VQ_fk_nps based on the framework in [24] is composed of four main phases: initialization phase, cluster update phase, incomplete data update phase, and termination phase.

In the initialization phase, we replace all missing values with the attribute means in the complete data subspace so that the initial clusters can be generated by vector quantization with the corresponding data vectors which are now complete. Phase 2 is a cluster update phase where the clustering procedure is run in the feature space to find the clusters with non-linear boundaries in the data space. This phase is mainly employed from [10] where the mean vectors are not explicitly computed. Instead, the clustering procedure keeps track of their members by means of a distance matrix d ($k \times n$ where k is the number of clusters and n is the number of objects). Different from an entire distance update on the distance matrix in [10], our work defines an incremental distance update on an individual cell of the distance matrix. As we check the convergence in step 2.2 right after the update of the clusters in the feature space, the convergence of the clustering procedure is preserved and guaranteed in our algorithm. It is based on the stability of the resulting clusters reflected by the stability of the distance matrix in the feature space. In phase 3, we handle incomplete data directly in an iterative way using their clusters in the data space each of which includes the similar data vectors. Thus, our scheme can conduct incomplete data handling towards the final clusters. In phase 4, we prepare for termination of the algorithm by returning the resulting clusters as several groups of the similar data vectors that are now completed.

Our incomplete educational data clustering algorithm, VQ_fk_nps, is defined as:

Input:

- ***D***: an input data set where incomplete data are present. There are n data vectors each of which is $X_j = (x_{j,1}, x_{j,2}, \ldots, x_{j,p})$ for j=1..n and p is the number of dimensions in the data space.
- σ: the width of the Gaussian kernel function K
- k: the number of clusters
- lr_0: an initial learning rate which is a positive value (lr_0<1.0); used to define the learning rate lr according to the function: $lr = lr_0$/the current number of *iterations*
- *threshold*: a value for the stopping criterion

Output:

- ***C***: a set of the k resulting clusters each of which has a reference vector $C_i = (c_{i,1}, c_{i,2}, \ldots, c_{i,p})$ where i = 1..k and p is the number of dimensions in the data space. In the feature space, k clusters are embodied in the distance matrix d (kxn) with k rows and n columns. Each cell in the matrix d is a distance between C_i and a data vector X_j in the feature space.
- ***D***: the input data set that has incomplete data imputed

Algorithm:

1. Initialization phase

1.1. Fill the incomplete data in ***D*** using the attribute means (am) in the complete data subspace: $am_1, am_2, \ldots, am_p$.

For each vector $X_j = (x_{j,1}, x_{j,2}, \ldots, x_{j,p})$ in ***D*** for j=1..n

Update the incomplete value at each dimension *dim* for *dim*=1..p with the value of the attribute mean at the same dimension using the incomplete data update rule:

$$x_{j,dim} = am_{dim}. \quad (1)$$

1.2. Initialize k reference vectors each of which C_i is a reference vector of a corresponding cluster obtained by roughly clustering the data vectors with vector quantization in the complete data space.

1.3. Initialize the distance matrix d in the feature space using the Gaussian kernel function K for each cell in the matrix as follows:

$$d_{ij} = 2*(1-K(C_i, X_j)) \text{ for } i=1..k \text{ and } j=1..n. \quad (2)$$

2. Cluster update phase

2.1. Update each cluster in the feature space based on the membership of each data vector in ***D*** by updating the distance matrix d.
For each data vector $X_j = (x_{j,1}, x_{j,2}, \ldots, x_{j,p})$ in ***D***:
2.1.1. Find the cluster that X_j belongs to based on the nearest distance between X_j and its corresponding reference vector C_l in the distance matrix d in the feature space.

$$C_l = \text{argmin}_i\, d_{ij}\,(C_i, X_j) \text{ for } i = 1..k \quad (3)$$

2.1.2. Update the distance between the vector X_j and the corresponding reference vector C_l to favor the membership of the vector X_j as follows:

$$d_{lj} = (1\text{-}lr)*d_{lj}^{\text{prev}} - lr*(1\text{-}lr)*d_{ll}^{\text{prev}} + 2*lr*(1\text{-}K(X_j,X_l)) \quad (4)$$

Where d_{lj}^{prev} and d_{ll}^{prev} are the previous distance between the reference vector C_l and the vector X_j and the one between the reference vector C_l and the vector X_l, respectively, in the previous iteration. The update of the distance d_{lj} is made to move the reference vector C_l closer to its member in the feature space while there is no change on the reference vectors of the other clusters. More details about this update of d_{lj} can be found in [10].
2.2. If the distance matrix d in the feature space is not stable with respect to *threshold*, then go to phase 3.

3. Incomplete data update phase

For each vector $X_j = (x_{j,1}, x_{j,2}, \ldots, x_{j,p})$ in ***D***:
3.1. Find the nearest cluster whose reference vector $C^* = (c^*_1, c^*_2, \ldots, c^*_p)$ in the data space based on the minimum distance in the distance matrix in the feature space where C^* is obtained as a mean vector in the complete data space using all the data vectors belonging to the same cluster in the feature space as the vector X_j.

$$C^* = \text{argmin}_i\, d_{ij}(C_i, X_j) \text{ for } i = 1..k. \quad (5)$$

3.2. Update the incomplete value at each dimension *dim* for the vector X_j with the value of the nearest reference vector C^* at the same dimension using the incomplete data update rule in the nearest prototype strategy:

$$x_{j,dim} = c^*_{dim}. \quad (6)$$

3.3. Update the learning rate: $lr = lr_0$/the current number of *iteration*s
3.4. Go back to phase 2.

4. Termination phase

4.1. Derive k clusters based on the distance matrix between the reference vectors of the clusters and the data vectors.
4.2. Return ***D*** with no more incomplete data.

An evaluation of the proposed approach from the theoretical perspectives

In this subsection, we would like to highlight the properties of our proposed approach from the theoretical perspectives as follows:

- Regarding the complexity of the proposed algorithm, an incremental update is performed to adjust the distance d_{1j} in the distance matrix with respect to the best matching between a current data vector X_j and its corresponding reference vector C_1 as specified in (4) at step 2.1.2 of the cluster update phase. This update follows a stochastic gradient descent instead of standard gradient descent which allows an entire update on the distance matrix after all the objects are assigned to their appropriate clusters. As a result, our learning process carries out an incremental update at a finer and faster pace as soon as we found the correct cluster of each object in each epoch. This might avoid stepping over the actual converging point and increasing the computational cost of cluster formulation through updating the distance matrix. Indeed, in

our algorithm, the total number of distance updates in each epoch is n because there is one update for the cluster assignment of each object. This is quite efficient as compared to the total number of distance updates on the entire distance matrix in each epoch which is $k \times n$.

- Reducing the randomness in initialization by means of the clustering procedure with vector quantization is considered in the initialization phase of our proposed algorithm. Consequently, our initial reference vectors in the data space are the resulting reference vectors from vector quantization in the complete data space. Such an initialization helps stabilizing the convergence of VQ_fk_nps speedily.
- In our opinion, the incomplete data update approach in the nearest prototype strategy is a special case of that in the optimal completion strategy regardless of fuzziness for the membership of each vector. Therefore, for incomplete data update, we prefer the nearest prototype strategy for hard clustering to the optimal completion strategy for fuzzy clustering. The reason from the practical point of view is that in our education clustering task, we need to distinguish the in-trouble students from the others so that appropriate support can be provided to the students. In addition, the employed clustering algorithm which is kernel-based vector quantization belongs to the category of hard clustering algorithms. Thus, our choice of the nearest prototype strategy is made appropriately.
- As presented previously, our VQ_fk_nps algorithm has an important inheritance from kernel-based vector quantization to reach the resulting non-spherical clusters in the data space in a simple, intuitive, and efficient learning scheme. Above all, there is no need of a concrete explicit mapping between the data space and the feature space. This leads to no need of any explicitly transformed version of either data vectors or their corresponding clusters while the clustering procedure iteratively formulates the clusters of the similar data vectors in the feature space. In order to remain the effectiveness and efficiency of kernel-based vector quantization, VQ_fk_nps has kept its spirit in tact in each epoch and simply enhanced the clustering procedure with the aforementioned incomplete data handling mechanism. Thus, VQ_fk_nps can be regarded as an incomplete data clustering version of kernel-based vector quantization.

3 Experimental results

For further evaluation from the empirical perspectives, we present several experimental results in this section. The experiments were performed on educational data including study results of the undergraduate students enrolled in 2005–2008 following the program in Computer Science in the academic credit system at Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology, Vietnam, [3]. Three data sets were prepared corresponding to 3 years of study from year 2 to year 4: data set "Year 2" for second-year students, "Year 3" for third-year students, and "Year 4" for fourth-year students. Each data set includes the study results of 1334 students (i.e., $n = 1334$). Each vector of a data set representing a student has 43 dimensions (i.e., $p = 43$) corresponding to 43 subjects. Missing grades appear corresponding to incomplete subjects as detailed in Table 1 where the percentage of missingness in data set "Year 2" is the highest and the one in "Year 4" is the lowest.

Table 1 Distribution of incomplete subjects over study years

Year of study		Year 2	Year 3	Year 4
Incomplete subjects	Total number	28,874	18,225	11,553
	Percentage (%)	50.34	31.77	20.14

As for the algorithms, we used the following algorithms for comparison with our proposed algorithm, VQ_fk_nps:

- **K_an** K_an is an algorithm that handles incomplete data in the data preprocessing phase by replacing incomplete data with the mean value at each dimension and then uses k-means [17] on the resulting data set.
- **VQ_an** VQ_an is an algorithm similar to K_an using vector quantization instead of k-means.
- **ImpSOM** ImpSOM is based on the self-organization map learning algorithm [14]. It tackles data incompleteness in the post-processing phase by means of the mean value at each dimension of each cluster for missing data in the same cluster after performing the self-organizing map on the incomplete data set in the complete data subspace. This algorithm is adapted from one in [7].
- **NPSFCM** NPSFCM is the algorithm proposed in [9] using the nearest prototype strategy for updating incomplete data while data clustering in the entire data space.
- **OCSFCM** OCSFCM is the algorithm proposed in [9] using the optimal completion strategy for incomplete data updates while clustering in the data space.
- **rmVQ_fk_nps** a variant of VQ_fk_nps is obtained by using random vectors in step 1.2 of the initialization phase as originally defined in vector quantization and using an entire update on the distance matrix after the cluster assignment of all the data vectors in step 2.1 of the cluster update phase as originally proposed in kernel-based vector quantization in [10]. We include this variant in order to check if our improvement on both vector quantization and kernel-based vector quantization is empirically appropriate.
- **mVQ_fk_nps** also as a variant of VQ_fk_nps, mVQ_fk_nps is obtained by using an entire update on the distance matrix after the cluster assignment of all the

Table 2 Averaged results of 30 runs from each algorithm

Algorithm	Xie_Beni			S_Dbw		
	Year 2	Year 3	Year 4	Year 2	Year 3	Year 4
K_an	1.11*	0.93*	1.24	0.57*	0.54*	0.46*
S_an	0.68*	1.2*	1.01	0.56*	0.56*	0.49*
VQ_an	0.68*	1.17*	1.12	0.56*	0.56*	0.49*
ImpSOM	0.58	1.07*	1.33	0.51*	0.44	0.45*
NPSFCM	2.36*	1.7*	2.07*	0.52*	0.46*	0.47*
OCSFCM	0.98*	0.96*	1.1	0.47*	0.48*	0.45*
rmVQ_fk_nps	0.83*	2.34*	6*	0.38*	0.48*	0.54*
mVQ_fk_nps	1.68*	2.73*	5.21*	0.4*	0.54*	0.59*
rVQ_fk_nps	0.91*	1.63*	5.13*	0.43*	0.48*	0.53*
kVQ_fk_nps	0.53	0.88	0.96	0.33	0.46*	0.38
VQ_fk_nps	0.31	0.56	0.82	0.33	0.43	0.37

* Average values with significance level at 0.05 as compared with VQ_fk_nps

data vectors in step 2.1 of the cluster update phase as originally proposed in kernel-based vector quantization in [10]. Different from rmVQ_fk_nps, we include the VQ-based initialization phase for this variant in order to check if our improvement on kernel-based vector quantization is empirically appropriate.

- **rVQ_fk_nps** a variant of VQ_fk_nps is obtained by using random vectors in step 1.2 of the initialization phase as originally defined in vector quantization. We include this variant in order to check if our initialization is empirically appropriate for reducing the unstability of the converged clusters stemming from random initialization in various runs of the kernel-based vector quantization procedure.
- **kVQ_fk_nps** similarly, a variant of VQ_fk_nps is obtained by using mean vectors from the k-means algorithm in step 1.2 of the initialization phase. This variant will help us figure out the appropriateness of our choice of vector quantization to determine the starting point of our clusters in the feature space.

As an extended version of the work in [24], we re-used the choices of the parameter values for NPSFCM and OCSFCM where the number c (or k) of clusters is 5 and the weighting exponent m is 1.25. As for other parameter values of an initial learning rate lr_0 and the bandwidth σ, we used a trial-and-error scheme which is quite tedious and thus, they will be automatically determined in our future work. The value of lr_0 is 0.9 and the value of σ for "Year 2" data set is 1.15, σ for "Year 3" data set is 2.11, and σ for "Year 4" data set is 2.02. In addition, the stopping criterion is set for all the experiments based on the stability of the resulting clusters or the distance matrix in kernel-based vector quantization which is exactly the same as one in [9], which is 10^{-5}. In order to avoid randomness in initialization, each experimental result for comparison in Table 2 is an averaged value calculated from 30 runs of each algorithm. Their corresponding standard deviations are given in Table 3. Besides, several results for K_an, ImpSOM, NPSFCM, and OCSFCM in Table 2 are inherited from the corresponding results of our previous work in [24].

Regarding examining how well VQ_fk_nps clusters incomplete educational data, we also used two internal clustering validation measures well-known in unsupervised learning: Xie_Beni and S_Dbw. Examined in [19], Xie_Beni is popularly used for fuzzy cluster validity with the inter-cluster separation defined as the minimum square distance between cluster centers and the intra-cluster compactness defined as the mean square distance between each object and its cluster center. The smaller Xie_Beni, the better the resultant clusters are. As analyzed in [16], S_Dbw is a measure that can examine the separation and compactness of the resulting clusters with respect to monotonicity, noise, density, subclusters, and skewed distributions in data. The smaller S_Dbw, the better the clusters are. More details about Xie_Beni and S_Dbw can be found in [16]. In addition, One-Way ANOVA was conducted with equal variances assumed for post hoc multiple comparisons with Bonferroni at the 0.05 level of significance. Levene Statistic is also included for a test of homogeneity of variances. In Table 2, a star (*) is used to denote the mean difference significant at the 0.05 level in comparison with the averaged results of VQ_fk_nps.

Presented in Table 2, the values of Xie_Beni and S_Dbw from our algorithm, VQ_fk_nps, are always the smallest ones with a very much difference from the results of the other algorithms. In particular, the difference between the fuzzy approaches and ours can be explained in such a way that our educational data sets have many distinct groups of similar students and thus, a choice of the nearest prototype strategy is appropriate. This result is also consistent with such a fact that we have got a small value (1.25) for the

Table 3 Standard deviations corresponding to the averaged results of 30 runs from each algorithm in Table 2

Algorithm	Xie_Beni			S_Dbw		
	Year 2	Year 3	Year 4	Year 2	Year 3	Year 4
K_an	0.22	0.34	0.24	0.19	0.01	0.03
S_an	≈0	≈0	≈0	≈0	≈0	≈0
VQ_an	≈0	≈0	≈0	≈0	≈0	≈0
ImpSOM	0.26	≈0	≈0	0.09	≈0	≈0
NPSFCM	0.07	≈0	0.33	≈0	≈0	≈0
OCSFCM	0.31	0.38	0.59	0.08	0.06	0.07
rmVQ_fk_nps	0.85	1.7	5.34	0.04	0.09	0.12
mVQ_fk_nps	2.39	2.67	2.97	0.04	0.11	0.17
rVQ_fk_nps	0.91	1.2	3.62	0.06	0.04	0.12
kVQ_fk_nps	0.41	0.4	0.22	0.02	0.05	0.03
VQ_fk_nps	0.02	0.12	0.05	0.02	0.02	0.03

weighting exponent m in [24]. Considering the variance at each individual measure, we observe a quite large range for Xie_Beni and a small one for S_Dbw from different approaches and algorithms except for VQ_fk_nps. Xie_Beni shows the clearer distinction between the algorithms while S_Dbw shows that distinction a little. Almost the differences between VQ_fk_nps and the other approaches are significant through a statistical test at the 0.05 level. Therefore, it can be concluded that VQ_fk_nps is suitable for educational data sets as compared to the others which have been examined. Besides, Table 3 gives us the standard deviations corresponding to the averaged results of 30 runs from each algorithm in Table 2. Via the values of those standard deviations, there is no much variance in the results from many various executions of VQ_fk_nps as compared to other variants of VQ_fk_nps, leading to the fact that VQ_fk_nps is more stable than its variants. Nevertheless, the most stable algorithms with almost no variance are S_an and VQ_an. Such a stability of S_an and VQ_an stems from the nature of the self-organizing learning process. As our algorithm performs the clustering of data vectors in the feature space and tackles the incomplete data handling in the data space, its stability might be influenced although we considered this feature by reducing the randomness in initialization.

As compared to the resulting clusters from the variants of VQ_fk_nps, the resulting clusters from VQ_fk_nps are more compact and separate from each other. In addition, all the differences between these variants except for kVQ_fk_nps are statistically significant. Such differences prove that the design of our proposed algorithm, VQ_fk_nps, is empirically sound. Regarding the statistically insignificant difference from the results from kVQ_fk_nps, it is understood that the initialization with k-means and the one with vector quantization are quite similar to each other. It seems to be certain as the resulting clusters from both K_an and VQ_an have the similar compactness and separation via the values of Xie_Beni and S_Dbw measures. Hence, stabilizing the initial clusters instead of random data vectors for the initial clusters helps us to obtain the clusters of better quality. Regarding the appropriateness of a distance update scheme in the feature space, VQ_fk_nps much outperforms rmVQ_fk_nps and mVQ_fk_nps. So, our choice of an incremental distance update scheme on only the distance associated with the winning cluster is confirmed to be more appropriate than a standard distance update scheme on an entire distance matrix.

Generally speaking, VQ_fk_nps can perform the data clustering task effectively on incomplete educational data and produce the non-spherical clusters with higher compactness and better separation in the data space through the internal clustering validation measures such as Xie_Beni and S_Dbw. As VQ_fk_nps always has the lowest values for Xie_Beni and S_Dbw, such experimental results have confirmed the robustness and effectiveness of VQ_fk_nps, for handling different amounts of incomplete data in educational data sets. Using the resulting clusters, several concrete groups of the 2nd-year students (or 3rd-year students or 4th-year students) at the same level of study performance can be determined and our support can be planned and provided appropriately for each group of the similar students. Besides, the resulting completed data can play a role of a good input for other mining tasks including educational data classification and association analysis.

4 Related works

For further comparison with the related works, this section examines several existing works such as [1,2,7–9,23,25,27] in incomplete data clustering. Among these works, Refs. [1,2,9,23] updated incomplete data while doing data clustering, Wang [25] translated incomplete data into so-called fuzzy observations before generating clusters, and Cottrell

and Letrémy [7] estimated incomplete data after attaining clusters.

As one of the first works, Hathaway and Bezdek [9] developed four different strategies handling incomplete data in fuzzy clustering: the whole data strategy (WDS), the partial distance strategy (PDS), the optimal completion strategy (OCS), the nearest prototype strategy (NPS) similar to OCS except for using the nearest prototypes. As a kernel-based extension to OCSFCM, a kernel-based fuzzy c-means algorithm (KFCM) was proposed in [27] using a kernel-induced metric in the data space instead of the conventional Euclidean metric and the optimal completion strategy for incomplete data handling. In comparison, VQ_fk_nps is different from NPSFCM and OCSFCM in [9] and KFCM in [27] in the following aspects. In Initialization phase, we suggested to use a mean of all the known values at each dimension to fill in incomplete data in VQ_fk_nps while OCSFCM and NPSFCM algorithms had no mention of such a particular initialization. In addition, VQ_fk_nps uses a kernel-based approach to produce the clusters with arbitrary shapes while those algorithms used a partitioning-based approach forming only the hyper-spherical shapes of the resulting clusters. Based on OCS and fuzzy SOM algorithm, Abidi and Yahia [2] proposed OCS-FSOM and an extension of OCS-FSOM called Multi-OCSFSOM. In these algorithms, a learning rate is used as a fuzzy membership value of the current input vector in the output cluster. Besides, the algorithms in [2] require the maximal iteration number for the learning process. This might lead to an early convergence. Differently, our work does not fix the number of iterations.

Wang [25] is another work based on SOM algorithm by transforming incomplete data into so-called fuzzy observations. This approach depended on the domains at the dimensions where missing values exist and a large number of fuzzy observations would be generated for each input. Based on SOM, Cottrell and Letrémy [7] clustered incomplete data in the complete data subspace by ignoring the dimensions with missing values during the learning process. After the learning process, missing value estimation is performed using the (weighted) mean values of the class of each vector. For a comparison with Cottrell and Letrémy [7], we adapted this approach to obtain a simplified algorithm ImpSOM in the previous section. However, Cottrell and Letrémy [7] might lose the details of the vectors to be clustered if the number of dimensions with missing values gets larger. Moreover, the missing data imputation was not tightly involved in the cluster forming process.

As one of the most recent works, Vatanen [23] has proposed a revision of handling missing values to the batch SOM algorithm which is called Imputation SOM. Unlike Refs. [7,23] did not ignore the missing values in the learning process. Indeed, Vatanen [23] used the current value of the corresponding prototype, i.e. a mean value, to fill each missing component of a vector. To some extent, the missing value updates in [23] are similar to that of VQ_fk_nps. Differently, instead of using the current prototype, VQ_fk_nps uses the nearest prototype of each vector after the update of all reference vectors via the update of the distance matrix has been done in the feature space. In [8], a SOM-based method has been introduced for data imputation in incomplete data matrices. Although not focusing on the resulting clusters, Folguera et al. [8] is somewhat related to a SOM-based incomplete data clustering approach. Their method needs a part of complete data vectors to pre-train a SOM and then impute incomplete data. Thus, Folguera et al. [8] is not applicable to our work because the educational application domain has the data sets where there is no such a set of complete data vectors from the 2nd-year, 3rd-year, and 4th-year students in an academic credit system. Using the mean shift algorithm as introduced in [6], AbdAllah and Shimshoni [1] considered handling incomplete data by means of so-called MD_E distance; but did not supply any incomplete data handling scheme.

Finally, among a large number of the existing works, it is hard to foresee which incomplete data handling techniques are certainly appropriate for a particular domain and also work well with any existing unsupervised learning algorithms. So, a study of handling incomplete data in a clustering task is needed for obtaining an effective cluster model in general and in the education domain particularly. In addition, VQ_fk_nps has the merit of discovering the non-spherical clusters of better quality in the incomplete educational data sets by means of a kernel-based clustering process in the feature space. This approach has not yet been supported by any existing works.

5 Conclusion

For educational decision making support, we would like to early detect and support the in-trouble students who have just spent two years, three years, or four years studying in an academic credit system. This situation asks us to collect their study results as soon as possible, leading to data incompleteness that is one of the troublesome characteristics of the data in analysis and mining tasks. Therefore, our work has to deal with a so-called incomplete educational data clustering task to discover some groups of the similar students based on their study performance at different points in study time. As a solution to the task, a robust and effective clustering approach based on kernel-based vector quantization is determined along with the nearest prototype strategy. The resulting clustering algorithm, VQ_fk_nps, is elaborated and discussed in comparison with several different approaches based on k-means, SOM, fuzzy c-means, kernel fuzzy c-means, etc. Different from the existing general-purpose solutions, VQ_fk_nps is more effective on the real

educational data sets at various levels of data incompleteness via internal clustering validation. It can generate the non-spherical clusters based on the clusters resulted in the feature space by a kernel-based vector quantization approach. In addition, the data sets which got completed after clustered can be utilized in other mining tasks.

As of this moment, we focus on which incomplete data clustering approach is appropriate for our educational domain by making kernel-based vector quantization robust and effective. In the future, we will further evaluate our approach for a more diversity of complete data sets. It is also interesting to consider obtaining a parameter-free kernel-based vector quantization approach whose parameter values are automatically derived from the inherent characteristics of each data set.

Acknowledgments This paper is funded by Ho Chi Minh City University of Technology, Vietnam National University at Ho Chi Minh City, under the Grant number T-KHMT-2015-27.

References

1. AbdAllah, L., Shimshoni, I.: Mean shift clustering algorithm for data with missing values. In: Proceedings of DAWAK, pp. 426-438 (2014)
2. Abidi, B., Yahia, S.B.: A new algorithm for fuzzy clustering handling incomplete dataset. Int. J. Artif. Intell. Tools **23**(4), 1–21 (2014)
3. Academic Affairs Office, Ho Chi Minh City University of Technology, Vietnam, http://www.aao.hcmut.edu.vn/dhcq.html (2014)
4. Bogarín, A., Romero, C., Cerezo, R., Sánchez-Santillán, M.: Clustering for improving educational process mining. In: Proceedings of LAK'14, pp. 1–5 (2014)
5. Campagni, R., Merlini, D., Verri, M.C.: Finding regularities in courses evaluation with k-means clustering. In: Proceedings of the 6th International Conference on Computer Supported Education, pp. 26–33 (2014)
6. Comaniciu, D., Meer, P.: Mean shift: a robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell. **24**(5), 603–619 (2002)
7. Cottrell, M., Letrémy, P.: Missing values: processing with the Kohonen algorithm. In: Proceedings of applied stochastic models and data analysis, pp. 489–496 (2005)
8. Folguera, L., Zupan, J., Cicerone, D., Magallanes, J.F.: Self-organizing maps for imputation of missing data in incomplete data matrices. Chemometr. Intell. Lab. Syst. **143**, 146–151 (2015)
9. Hathaway, R.J., Bezdek, J.C.: Fuzzy c-means clustering of incomplete data. IEEE Trans. Syst. Man Cybernet. Part B Cybernet. **31**(5), 735–744 (2001)
10. Inokuchi, R., Miyamoto, S.: LVQ clustering and SOM using a kernel function. In: Proceedings of the 2004 IEEE International Conference on Fuzzy Systems, vol. 3, pp. 1497–1500 (2004)
11. Inyang, U.G., Joshua, E.E.: Fuzzy clustering of students' data repository for at-risks students identification and monitoring. Comput. Inf. Sci. **6**(4), 37–50 (2013)
12. Jayabal, Y., Ramanathan, C.: Clustering students based on student's performance—a partial least squares path modeling (PLS-PM) study. In: Proceedings of MLDM, LNAI 8556, pp. 393–407 (2014)
13. Kerr, D., Chung, G.K.W.K.: Identifying key features of student performance in educational video games and simulations through cluster analysis. J. Educ. Data Min. **4**(1), 144–182 (2012)
14. Kohonen, T.: The self-organizing map. Proc. IEEE **78**(9), 1464–1480 (1990)
15. Li, C., Yoo, J.: Modeling student online learning using clustering. In: Proceedings of ACM SE'06, pp. 1–6 (2006)
16. Liu, Y., Li, Z., Xiong, H., Gao, X., Wu, J.: Understanding of internal clustering validation measures. In: Proceedings of the 2010 IEEE International Conference on Data Mining, pp. 911–916 (2010)
17. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the 5th Berkeley Symp. Math. Stat. Prob., vol. 1, pp. 281–297 (1967)
18. Nugent, R., Dean, N., Ayers, E.: Skill set profile clustering: the empty k-means algorithm with automatic specification of starting cluster centers. In: Proceedings of the 3rd International Conference on Educational Data Mining, pp. 151–160 (2010)
19. Pal, N.R., Bezdek, J.C.: On cluster validity for the fuzzy c-means model. IEEE Trans. Fuzzy Syst. **3**(3), 370–379 (1995)
20. Pardos, Z.A., Trivedi, S., Heffernan, N.T., Sárközy, G.N.: Clustered knowledge tracing. In: Proceedings of ITS, LNCS 7315, pp. 405–410 (2012)
21. Shih, B., Koedinger, K.R., Scheines, R.: Unsupervised discovery of student learning tactics. In: Proceedings of the 3rd International Conference on Educational Data Mining, pp. 201–210 (2010)
22. Tanai, M., Kim, J., Chang, J.H.: Model-based clustering analysis of student data. In: Proceedings of ICHIT 2011, LNCS 6935, pp. 669–676 (2011)
23. Vatanen, T., Osmala, M., Raiko, T., Lagus, K., Sysi-Aho, M., Orešič, M., Honkela, T., Lähdesmäki, H.: Self-organization and missing values in SOM and GTM. Neurocomputing **147**, 60–70 (2015)
24. Vo, T.N.C., Nguyen, H.P., Vo, T.N.T.: A robust and effective algorithmic framework for incomplete educational data clustering. In: Proceedings of the 2nd National Foundation for Science and Technology Development Conference on Information and Computer Science (NICS), pp. 65–70 (2015)
25. Wang, S.: Application of self-organising maps for data mining with incomplete data sets. Neural Comput. Appl. **12**, 42–48 (2003)
26. Zakrzewska, D.: Cluster analysis in personalized e-learning systems. Intel. Syst. Knowl. Manag. SCI **252**, 229–250 (2009)
27. Zhang, D.-Q., Chen, S.-C.: Clustering incomplete data using kernel-based fuzzy c-means algorithm. Neural Process. Lett. **18**, 155–162 (2003)